REMARKS

The Office Action dated July 15, 2004 has been received and carefully noted. The above amendments to the claims, and the following remarks, are submitted as a full and complete response thereto.

Claims 7, 11, 15, and 21 have been amended to more particularly point out and distinctly claim the subject matter of the invention. No new matter has been added. Claims 1-25 are pending in the application and are respectfully submitted for consideration.

The Office Action stated that the information disclosure statement filed on June 5, 2002 fails to comply with 37 C.F.R. §1.98(a)(2), which requires a legible copy of each U.S. and foreign patent, and each publication. Copies of the requested documents are submitted along with this Response.

Claims 11 and 15 were rejected under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter of the invention. Specifically, the Office Action stated that there is insufficient antecedent basis for "the snooping module" and "the snooping means" in claims 11 and 15, respectively. Claims 11 and 15 have been appropriately amended and therefore the rejection under 35 U.S.C. §112, second paragraph, is rendered moot.

Claims 1-4, 6-9, 11-14, and 19 were rejected under 35 U.S.C. §102(b) as being anticipated by Kalkunte (U.S. Patent No. 6,031,821). The above rejection is respectfully traversed for the reasons which follow.

Claim 1, upon which claims 2-5 are dependent, recites a method of controlling data flow within a network device. The method includes the steps of receiving a data packet into the network device, snooping the data packet before the data packet is stored in a memory buffer of the network device to determine a packet size based upon a number of bits per bytes within the data packet, aggregating the packet size to generate a total number of data packets within a burst if the packet size exceeds a predetermined packet size, lowering a threshold of the memory buffer to a reset threshold if the total number of data packets exceeds a predetermined number of consecutive data packets, and activating a pause frame based upon the reset threshold to temporarily suspend transmission of incoming data packets to the network device.

Claim 6, upon which claims 8-10 are dependent, recites a device for controlling data flow within a network device. The device includes a snooping module contained within the network device and configured to snoop a data packet before the data packet is stored in a memory buffer of the network device to determine a packet size based upon the bits per byte of the data packet, and a counter connected to the snooping module, wherein the counter adds the packet size to generate a total number of data packets within a burst if the packet size exceeds a predetermined packet size. The device further includes a threshold lowering module connected to receive instructions from the

snooping module and configured to lower a threshold of the memory buffer to a reset threshold if the total number of data packets exceeds a predetermined number of consecutive data packets, and a pause activation module configured to receive instructions from the threshold lowering module in order to trigger a pause frame based upon the reset threshold to temporarily suspend transmission of incoming data packets to the network device.

Claim 11, upon which claims 12-15 are dependent, recites a device for controlling data flow within a network device. The device includes receiving a data packet into the network device. The device further includes snooping means contained within the network device for snooping a data packet before the data packet is stored in a memory buffer of the network device to determine a packet size based upon a number of bits/bytes of the data packet, aggregating means included within the snooping means for aggregating the packet size to generate a total number of data packets within a burst if the packet size exceeds a predetermined packet size, threshold reset means connected to receive instructions from the snooping means for lowering a threshold of the memory buffer to a reset threshold if the total number of data packets exceeds a predetermined number of consecutive data packets, and pause frame activation means connected to receive instructions from the threshold lowering module for activating a pause frame based upon the reset threshold to temporarily suspend transmission of incoming data packets to the network device.

Claim 16, upon which claims 17 and 18 are dependent, recites a method of

controlling data flow within a multiple-linked chip device. The method includes the steps of receiving the data packet into the multiple-linked chip device, snooping data packets before the data packets are stored in a memory buffer of the multiple-linked chip device to determine a packet size based upon the bits per bytes of the data packets, snooping the data packets received at both an input port and an expansion port connected to the multiple-linked chip to determine a packet size, aggregating the packet size of the data packets to generate a total number of data packets within a burst if the data packet size exceed a predetermined packet size, lowering a threshold of the memory buffer to a reset threshold if the total number of data packets exceeds a predetermined number of consecutive data packets, and activating a pause frame based upon the reset threshold to temporarily suspend transmission of incoming data packets to the multiple-linked chip.

Claim 19, upon which claims 20-21 are dependent, recites a device for controlling data flow within a multiple-linked chip device. The device includes a receiving module for receiving the data flow within the multiple linked chip device, a snooping module contained within the multiple-linked chip device and configured to snoop data packets before the data packets are stored in a memory buffer of the network device to determine a packet size based upon the bits per bytes of the data packets, a counter included within the snooping module, wherein the counter adds packet size of the data packets to generate a total number of data packets within a burst if the packet size exceeds a predetermined packet size, a threshold lowering module connected to receive instructions from the snooping module and configured to lower a threshold of the memory buffer to a reset

threshold if the total number of data packets exceeds a predetermined number of consecutive data packets, and a pause activation module configured to receive instructions from the threshold lowering module in order to trigger a pause frame based upon the reset threshold to temporarily suspend transmission of incoming data packets to the multiple-linked chip device.

Claim 22, upon which claims 23-24 are dependent, recites a device for controlling data flow within a multiple-linked chip device. The device includes snooping means contained within the multiple-linked chip device for snooping data packets before the data packets are stored in a memory buffer of the multiple-linked chip device to determine a packet size. The snooping means snoops the data packets received at both an input port and an expansion port connected to the multiple-linked chip to determine a packet size of the data packets received at the input port and the expansion port. The device further includes aggregating means included within the snooping module for aggregating the packet size of the data packets to generate a total number of data packets within a burst if the data packet size exceed a predetermined packet size, threshold reset means connected to receive instructions from the snooping module for lowering a threshold of the memory buffer to a reset threshold if the total number of data packets exceeds a predetermined number of consecutive data packets, and pause frame activation means connected to receive instructions from the threshold lowering module for activating a pause frame based upon the reset threshold to temporarily suspend transmission of incoming data packets to the multiple-linked chip.

Claim 25 recites a method of controlling data flow within a network device. The method includes predicting a future flow of a chip located within the network device based upon a current flow within another chip and the current flow within the chip, and determining whether the future flow will cause a memory buffer of the chip to become saturated.

The cited prior art reference of Kalkunte fails to disclose or suggest all of the elements of the claims, and therefore fails to provide the features discussed above.

Kalkunte discloses an apparatus and method for generating a pause frame in a buffered distributor based on lengths of data packets distributed according to a round robin repeater arbitration. A buffered distributor determines a pause frame interval specifying a necessary time interval for eliminating a congestion condition in an identified one of the receive buffers, the pause frame interval determined based on the relative packet length of data packets output by the buffered distributor before eliminating the congestion condition according to a prescribed output logic, the output data rate of the buffered distributor, and the associated overhead delay in the buffered distributor.

Claim 1 of the claimed invention recites, in part, "snooping the data packet before the data packet is stored in a memory buffer of the network device to determine a packet size based upon a number of bits per bytes within the data packet." Kalkunte, on the other hand, only discloses monitoring the length of incoming data packets (Kalkunte, Column 5, lines10-12). Moreover, in Kalkunte, the receive FIFO buffer is the component

that counts the number of bytes in a received data packet (Kalkunte, Column 5, lines10-12). The FIFO buffer is contained within the network port, and therefore the length of the packet is not determined until after it enters the memory buffer of the network device.

In the claimed invention, on the other hand, the data packets are snooped before they are stored in a memory buffer of the network device. Packets entering through both the input ports and the expansion ports are snooped to determine if a burst of consecutive large data packets will be transmitted into the chip's memory buffer (Specification, Page 8, Paragraph 19). This is a significant improvement over that which is disclosed in Kalkunte. The claimed invention allows the system to look ahead and make appropriate determinations or changes before the data packet enters the buffer memory of the network device. Therefore, the claimed invention may take preemptive measures to prevent congestion of the memory buffer from occurring. Additionally, since the memory buffer will not be overflowed, incoming data packets will not be dropped when received into the buffer. In any case, Kalkunte does not disclose or suggest snooping the data packet before the data packet is stored in a memory buffer of the network device, and therefore does not provide the advantages of the claimed invention.

Kalkunte also fails to disclose or suggest aggregating the packet size to generate a total number of data packets within a burst if the packet size exceeds a predetermined packet size, and lowering the threshold of the memory buffer to a reset threshold if the total number of data packets exceeds a predetermined number of consecutive data packets. Rather, Kalkunte only discloses keeping an ordered list of all arrivals of data

packets in the congested receive FIFO (Kalkunte, Column 6, lines 61-66). Kalkunte fails to disclose determining whether an incoming data packet size exceeds a predetermined data packet size, adding an incoming data packet which exceeds the predetermined size to the total number of data packets within a burst, and then determining whether the total number of data packets within the burst exceeds a predetermined number of consecutive data packets within the burst (Specification, Page 13, Paragraph 26). The solution disclosed in Kalkunte is only designed to ensure that the entire FIFO buffer is empty before receiving another packet (Kalkunte, Column 6, lines 66-67). Kalkunte contains no disclosure or suggestion related to comparing incoming data packet sizes to a predetermined packet size, or comparing the total number of data packets within a burst to a predetermined number of consecutive data packets within the burst.

Consequently, Kalkunte fails to anticipate all of the elements of claim 1, and applicants respectfully request the withdrawal of the rejection. Applicants further submit that claims 2-5 should be allowed for at least their dependence upon claim 1, and for the specific limitations recited therein.

With respect to the rejection of claim 6, Kalkunte fails to disclose or suggest a snooping module configured to snoop a data packet before the data packet is stored in a memory buffer of the network device. The snooping module of the claimed invention, as discussed in the Specification, is configured to snoop the packets entering through both the input ports and the expansion ports to determine if a burst of consecutive large data packets will be transmitted into the memory buffer. The snooping module is capable of

looking ahead before the data packet enters the network device to find out whether successive large packets will be transferred into the device in order to determine in advance whether the memory buffer will become saturated upon receiving the incoming data packets. The snooping module is also capable of determining whether successive large packets received simultaneously at both the input and expansion ports will have an aggregate affect such that the memory buffer will become saturated (Specification, Page 8, Paragraph 19). No such snooping module is disclosed or suggested by Kalkunte.

Additionally, Kalkunte does not disclose or suggest a snooping module which will instruct a threshold lowering module to lower a threshold of the memory buffer to a reset threshold. If the snooping module determines that the burst of incoming data packets will most likely cause an overflow situation to occur, the snooping module will instruct a flow control module to lower the threshold of the memory buffer (Specification, Page 8, Paragraph 20).

Furthermore, the snooping module may be programmed to determine if a predetermined number of consecutive large packets are being transmitted into the network device. The system will then lower the threshold to activate a pause frame if the size of the incoming data packets and if the number of consecutive incoming packets exceeds a predetermined packet size. Also, the snooping module may be programmed to snoop for consecutive data packets having a predetermined transmission rate. If a predetermined number of consecutive data packets are received within the system, the snooping module will use this information to predict whether the receipt of additional

data packets will cause the memory buffer to become saturated (Specification, Page 10, Paragraph 22). Kalkunte fails to disclose or suggest such a snooping module.

Therefore, Kalkunte fails to disclose or suggest all of the elements of claim 6, and applicants respectfully request withdrawal of the rejection. Claims 7-10 should also be allowed for at least their dependence upon claim 6, and for the specific limitations recited therein.

Claim 11 contains certain similar limitations as those recited in claim 1. Kalkunte, as discussed above, does not disclose or suggest "snooping means for snooping a data packet before the data packet is stored in a memory buffer," nor does it disclose "aggregating means included within the snooping means for aggregating the packet size to generate a total number of data packets within a burst if the packet size exceeds a predetermined packet size." In addition, Kalkunte fails to disclose or suggest a snooping module which instructs a threshold reset means.

Therefore, Kalkunte does not disclose or suggest all of the elements of claim 11, and applicants respectfully request the withdrawal of the rejection. Claims 12-15 should also be allowed for at least their dependence upon claim 11, and for the specific limitations recited therein.

Claim 19 contains a similar limitation to claim 6, specifically a snooping module. Kalkunte, as discussed above, does not disclose or suggest a snooping module. Thus, Kalkunte fails to disclose or suggest all of the elements of claim 19. Applicants respectfully submit that claims 20 and 21 should also be allowed for at least their

dependence upon claim 19, and for the specific limitations recited therein.

Claims 5, 10, 15-18, and 20-24 were rejected under 35 U.S.C. §103(a) as being unpatentable over Kalkunte in view of Erimli (U.S. Patent No. 6,405,258). The Office Action stated that Kalkunte teaches all of the elements of the claims, with the exception of the step of snooping includes snooping the data packet at the input port and expansion port. The Office Action then relies on Erimli to cure the deficiency inherent in Kalkunte. The above rejection is respectfully traversed for the reasons which follow.

Kalkunte is discussed above. Erimli discloses a method and apparatus for controlling the flow of data frames through a network switch on a port-by-port basis. A receive port receives data frames from a first network station, and a transmit port outputs the received data frames to a second network station. A programmable threshold register is provided for storing a threshold value that indicates a saturation level for the internal resources of the transmit port. If the threshold value has been reached, then control circuitry will implement a flow control process that causes the first network station to discontinue transmission of data frames to the transmit port until the internal resources of the transmit port fall below the threshold value.

Claim 5 is dependent upon claim 1. Erimli fails to cure the deficiencies in Kalkunte, discussed above, with respect to claim 1. The combination of Erimli and Kalkunte fails to disclose or suggest "snooping the data packet before the data packet is stored in a memory buffer of the network device." Thus, the combination of Kalkunte and Erimli fail to disclose or suggest all of the elements of claim 5.

Claim 10 is dependent upon claim 6. Erimli fails to cure the deficiencies in Kalkunte, discussed above, with respect to claim 6. In other words, the combination of Erimli and Kalkunte also fails to disclose a snooping module. Thus, the combination of Kalkunte and Erimli fail to disclose or suggest all of the elements of claim 10.

Claim 15 is dependent upon claim 11. Erimli fails to cure the deficiencies in Kalkunte, discussed above, with respect to claim 11. The combination of Erimli and Kalkunte does not disclose or suggest snooping means or aggregating means, as recited in the claims. Thus, the combination of Kalkunte and Erimli fail to disclose or suggest all of the elements of claim 15.

With respect to the rejection of claim 16, Kalkunte, as stated above, fails to disclose or suggest snooping data packets before the data packets are stored in a memory buffer, or aggregating the packet size if the data packet size exceeds a predetermined packet size. Furthermore, Erimli also fails to disclose or suggest such a limitation and therefore fails to cure the deficiency in Kalkunte. Thus, Kalkunte and Erimli, whether taken alone or in combination, fail to disclose all of the elements of claim 16.

Applicants respectfully submit that claims 17 and 18, which depend upon claim 16, should be allowed for at least their dependence upon claim 16, and for the specific limitations recited therein.

Claims 20 and 21 are dependent upon claim 19. Erimli fails to cure the deficiencies in Kalkunte, discussed above, with respect to claim 19. Both Erimli and Kalkunte, whether taken alone or in combination, do not disclose or suggest a snooping

module. Thus, the combination of Kalkunte and Erimli fail to disclose or suggest all of the elements of claims 20 and 21.

With respect to the rejection of claim 22, Kalkunte, as stated above, fails to disclose or suggest snooping means or aggregating means. Erimli also fails to disclose or suggest such a limitation and therefore fails to cure the deficiency in Kalkunte. Thus, Kalkunte and Erimli, whether taken alone or in combination, fail to disclose all of the elements of claim 22. Applicants respectfully submit that claims 23 and 24 should also be allowed for their dependence upon claim 22, and for the specific limitations recited therein.

Applicants respectfully assert that the combination of Kalkunte and Erimli fail to disclose or suggest all of the elements of claims 5, 10, 15-18, and 20-24. Consequently, Applicants respectfully request the withdrawal of this rejection.

Claim 25 was rejected under 35 U.S.C. §103(a) as being unpatentable over Lam (U.S. Patent No. 6,553,027) in view of Erimli (U.S. Patent No. 6,405,258). The Office Action took the position that Lam teaches all of the elements of claim 25, with the exception of predicting a future flow of a chip based upon the current flow of the chip and another chip, and determining whether the future flow will cause a memory buffer chip to become saturated. The above rejection is respectfully traversed for the reasons which follow.

Erimli is discussed above. Lam discloses an apparatus and method for cascading multiple network switch devices. A network switch arrangement in a packet switched network connects a plurality of multiport network switches which is in a circular, serial manner so that data is transferable between the network switches only unidirectionally. When it is determined that data is to be transmitted from a first network switch to a port in a second network switch, the data is transmitted over an expansion bus from the first network switch to the second network switch in a continuous stream of data bursts.

The Office Action relies on Erimli for the disclosure of predicting a future flow of a chip based upon the current flow of the chip and another chip, and determining whether the future flow will cause a memory buffer chip to become saturated. Erimli, however, only discloses a CPU which monitors the flow of data through a multiport switch and programs the value of threshold registers based on the amount of data through each output port of the multiport switch (Erimli, Column 14, lines 32-37). Erimli does not disclose predicting a future flow as recited in the claim, rather Erimli discloses monitoring the present flow of data through the multiport switch and making present decisions based on that flow. Additionally, Erimli does not disclose monitoring the current flow within another chip and the current flow within the same chip.

Erimli also fails to disclose "determining whether the future flow will cause a memory buffer of the chip to become saturated," as recited in claim 25. Erimli specifically discloses that "upon detecting that output queue 58b has reached its threshold value, the control logic 96 is responsible for implementing a flow control

technique to prevent output queue 58b from becoming completely full" (Erimli, Column 15, lines 13-17). Thus, Erimli only begins to prevent saturation when the output queue has reached its threshold value. It is not predicting a future flow and then determining whether that future flow will cause the memory buffer to become saturated.

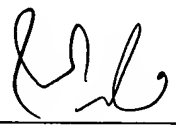
As a result, the combination of Lam and Erimli fails to disclose or suggest all of the elements of claim 25. For at least those reasons, Applicants respectfully request the withdrawal of the rejection of claim 25.

The cited references of Kalkunte, Erimli, and Lam, whether taken alone or in combination, fail to disclose or suggest critical and important elements of the claimed invention. These distinctions are more than sufficient to render the claimed invention unanticipated and unobvious. It is therefore respectfully requested that all of claims 1-25 be allowed, and this application passed to issue.

If for any reason the Examiner determines that the application is not now in condition for allowance, it is respectfully requested that the Examiner contact, by telephone, the applicants' undersigned attorney at the indicated telephone number to arrange for an interview to expedite the disposition of this application.

In the event this paper is not being timely filed, the applicants respectfully petition for an appropriate extension of time. Any fees for such an extension together with any additional fees may be charged to Counsel's Deposit Account 50-2222.

Respectfully submitted,



Majid S. AlBassam
Registration No. 54,749

Customer No. 32294
SQUIRE, SANDERS & DEMPSEY LLP
14TH Floor
8000 Towers Crescent Drive
Tysons Corner, Virginia 22182-2700
Telephone: 703-720-7800
Fax: 703-720-7802

MSA/tdg